In the Claims:

Please amend the claims as follows:

1. (currently amended) A peer-vector machine, comprising:

a host processor operable to execute an ~~program~~ instruction of a program, ~~and, in response to the program instruction, operable~~ to generate first host data in response to the instruction, and to identify configuration information that is separate from the program; and

a pipeline accelerator coupled to the host processor and operable to

receive the identified configuration information,

configure a portion of the pipeline accelerator in response to the configuration information,

receive the first host data, and

generate first pipeline data from the first host data without executing any program instruction.

2. (original) The peer-vector machine of claim 1 wherein the host processor is further operable to:

receive second data; and

generate the first host data from the second data.

3. (original) The peer-vector machine of claim 1 wherein the host processor is further operable to:

receive the first pipeline data from the pipeline accelerator; and

process the first pipeline data.

4. (original) The peer-vector machine of claim 1 wherein the host processor is further operable to:

receive the first pipeline data from the pipeline accelerator; and

generate the first host data from the first pipeline data.

5. (original) The peer-vector machine of claim 1, further comprising:

an interface memory coupled to the host processor and to the pipeline accelerator and having a first memory section;

wherein the host processor is operable to,

store the first host data in the first memory section, and

provide the first host data from the first memory section to the pipeline accelerator.

6. (original) The peer-vector machine of claim 1, further comprising:

an interface memory coupled to the host processor and to the pipeline accelerator and having first and second memory sections;

wherein the host processor is operable to,

store the first host data in the first memory section,

provide the first host data from the first memory section to the pipeline accelerator,

receive the first pipeline data from the pipeline accelerator,

store the first pipeline data in the second memory section,

retrieve the first pipeline data from the second memory section to the host processor, and

process the first pipeline data.

7. (currently amended) The peer-vector machine of claim 1 wherein the host processor is operable to ~~configure the pipeline accelerator~~provide the identified configuration information to the pipeline accelerator.

8. (currently amended) The peer-vector machine of claim 1 wherein:

the identified configuration information comprises firmware; and

-the portion of the pipeline accelerator comprises a programmable-logic integrated circuit that the firmware is operable to configure.

9. (currently amended) A peer-vector machine, comprising:

a pipeline accelerator operable to generate first pipeline data without executing any program instruction; and

a host processor coupled to the pipeline accelerator and operable to execute a program ~~instruction and~~, ~~in response to the program instruction,~~ operable to receive the first pipeline data and to generate first host data from the first pipeline data in response to the executing program, and, while not executing the program, operable to configure a portion of the pipeline accelerator to generate the first pipeline data.

10. (original) The peer-vector machine of claim 9 wherein the pipeline accelerator is further operable to:

receive second data; and

generate the first pipeline data from the second data.

11. (original) The peer-vector machine of claim 9 wherein the pipeline accelerator is further operable to:

receive the first host data from the host processor; and

process the first host data.

12. (original) The peer-vector machine of claim 9 wherein the pipeline accelerator is further operable to:

receive the first host data from the host processor; and

generate the first pipeline data from the first host data.

13. (original) The peer-vector machine of claim 9, further comprising:

an interface memory coupled to the pipeline accelerator and to the host processor and having a first memory section; and

wherein the host processor is operable to,

store the first pipeline data from the pipeline accelerator in the first memory section, and

retrieve the first pipeline data from the first memory section.

14. (original) The peer-vector machine of claim 9, further comprising:

an interface memory coupled to the pipeline accelerator and to the host processor and having first and second memory sections;

wherein the host processor is operable to,

store the first pipeline data from the pipeline accelerator in the first memory section,

retrieve the first pipeline data from the first section,

store the first host data in the second memory section, and

provide the first host data from the second memory section to the pipeline accelerator; and

wherein the pipeline accelerator is operable to process the first host data received from the second memory section.

15. Cancelled.

16. (currently amended) A system, comprising:

a device operable to generate raw data;

a host processor coupled to the device and operable to execute an instruction of a program~~ instruction~~, ~~and,~~ in response to the ~~program~~ instruction~~,~~ operable to generate host data from the raw data, and operable to identify configuration information that is separate from the program; and

a pipeline accelerator coupled to the host processor and operable to

receive the configuration information;

configure at least a portion of the pipeline accelerator in response to the configuration information;

receive the host data,

generate pipeline data from the host data without executing any program instruction; and

provide the pipeline data to a destination remote from the pipeline accelerator.

17. (currently amended) A system, comprising:

a device operable to generate raw data;

a pipeline accelerator coupled to the device and operable to generate pipeline data from the raw data without executing any program instruction; and

a host processor coupled to the pipeline accelerator and operable to execute a program~~ instruction and~~, in response to the program~~ instruction,~~ operable to receive the pipeline data and to generate host data from the pipeline data, and, while not executing the program, operable to configure a portion of the pipeline accelerator to generate the pipeline data.

18. (currently amended) A method, comprising:

before executing a first program with a host processor, configuring with the host processor a portion of a pipeline accelerator to generate first pipeline data from first host data;

generating the first host data by executing a first ~~program~~ instruction of the program with the~~a~~ host processor; and

generating the first pipeline data from the first host data with the configured~~a~~ pipeline accelerator without executing any program instruction.

19. (original) The method of claim 18, further comprising:

receiving raw data;

wherein generating the first host data comprises generating the first host data from the raw data.

20. (original) The method of claim 18 wherein generating the first host data comprises generating the first host data from the first pipeline data.

21. (currently amended) The method of claim 18, further comprising generating second host data from the first pipeline data by executing an instruction of the program ~~instruction~~ with the host processor.

22. (currently amended) The method of claim 18 wherein ~~, further comprising~~ configuring the pipeline accelerator comprises configuring a portion of the pipeline accelerator by executing an ~~program~~ instruction of a second program with the host processor.

23. (currently amended) A method, comprising:

identifying configuration information with a host processor;

configuring a portion of a pipeline accelerator with the identified configuration information;

generating first pipeline data with the configured~~a~~ pipeline accelerator without executing any program instruction; and

generating first host data from the first pipeline data by executing ~~a program instruction~~ with the ~~a~~ host processor an instruction of a first program that is separate from the identified configuration information.

24. (original) The method of claim 23, further comprising:

receiving raw data;

wherein generating the first pipeline data comprises generating the first pipeline data from the raw data.

25. (original) The method of claim 23 wherein generating the first pipeline data comprises generating the first pipeline data from the first host data.

26. (original) The method of claim 23, further comprising generating second pipeline data from the first host data with the pipeline accelerator.

27. (currently amended) The method of claim 23 wherein identifying the configuration information comprises identifying the configuration information~~, further comprising configuring the pipeline accelerator~~ by executing an ~~program~~ instruction of a second program with the host processor.

28. (currently amended) A peer-vector machine, comprising:

a host processor operable to execute an~~ program~~ instruction of a program, ~~and, in~~ to generate host data in response to the ~~program~~ instruction, and ~~operable~~ to identify configuration information that is separate from the program~~generate first host data~~; and

a pipeline accelerator coupled to the host processor and operable to

receive the identified configuration information,

configure a portion of the pipeline accelerator in response to the configuration information,

receive the ~~first~~ host data, and

generate ~~first~~ pipeline data from the ~~first~~ host data without responding to any program instruction.

29. (currently amended) A peer-vector machine, comprising:

a pipeline accelerator operable to

receive host data, and

generate pipeline data from the host data without receiving any program instruction; and

a host processor coupled to the pipeline accelerator and operable to execute a program~~ instruction~~, ~~and, in response to the program instruction,~~ operable to generate the ~~first~~ host data in response to the executing program, and, while not executing the program, operable to configure a portion of the pipeline accelerator to generate the pipeline data; and

~~a pipeline accelerator coupled to the host processor and operable to~~

~~receive the first host data, and~~

~~generate first pipeline data from the first host data without receiving any program instruction~~.

30. (new) The peer-vector machine of claim 1 wherein the pipeline accelerator is operable to configure the portion of the pipeline accelerator before generating the first pipeline data.

31. (new) The peer-vector machine of claim 1 wherein the pipeline accelerator is operable to configure the portion of the pipeline accelerator before receiving the first host data.

32. (new) The peer-vector machine of claim 1, further comprising:

an accelerator-configuration registry operable to store the configuration information; and

wherein the host processor is further operable to transfer the configuration information from the registry to the pipeline accelerator.

33. (new) The peer-vector machine of claim 1 wherein the host processor is operable to identify the configuration information before executing the instruction.

34. (new) The peer-vector machine of claim 9 wherein the host processor is operable to configure the portion of the pipeline accelerator before executing the program.

35. (new) The peer-vector machine of claim 9, further comprising:

an accelerator-configuration registry operable to store configuration information; and

wherein the host processor is operable to configure the portion of the pipeline accelerator by transferring the configuration information from the registry to the pipeline accelerator.